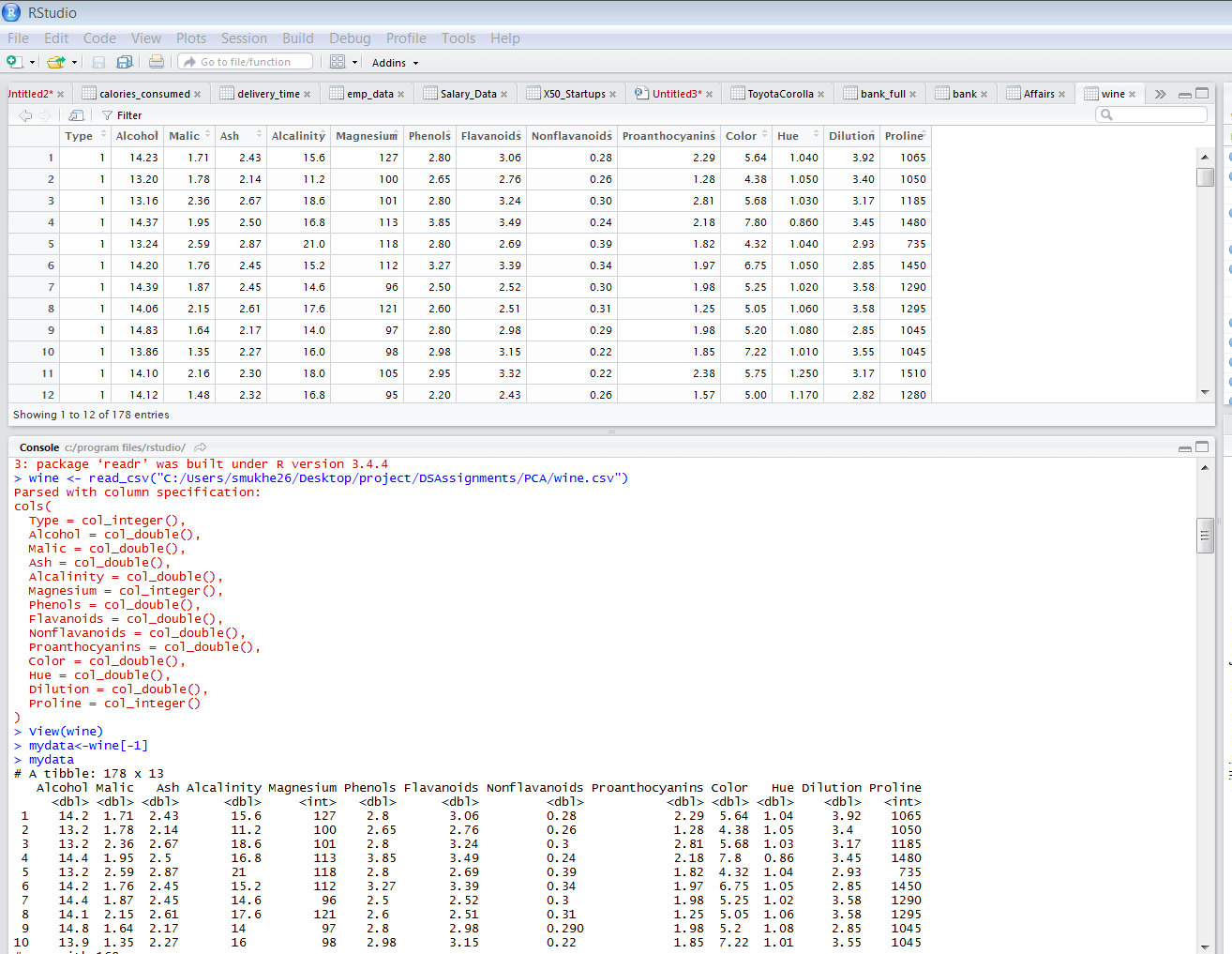
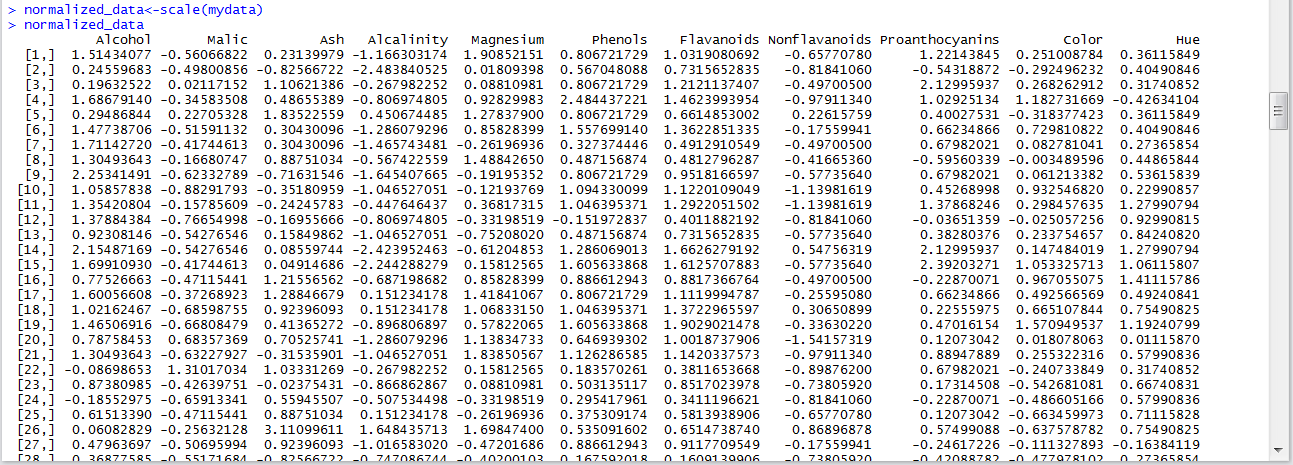
Perform Principal component analysis and perform clustering using first 3 principal component scores (both heirarchial and k mean clustering(scree plot or elbow curve) and obtain optimum number of clusters and check whether we have obtained same number of clusters with the original data (class column we have ignored at the begining who shows it has 3 clusters)df

Before applying PCA lets first try to solve without using it

Let’s import the wine.csv into R studio and remove the first column since it already shows which cluster the row belongs to



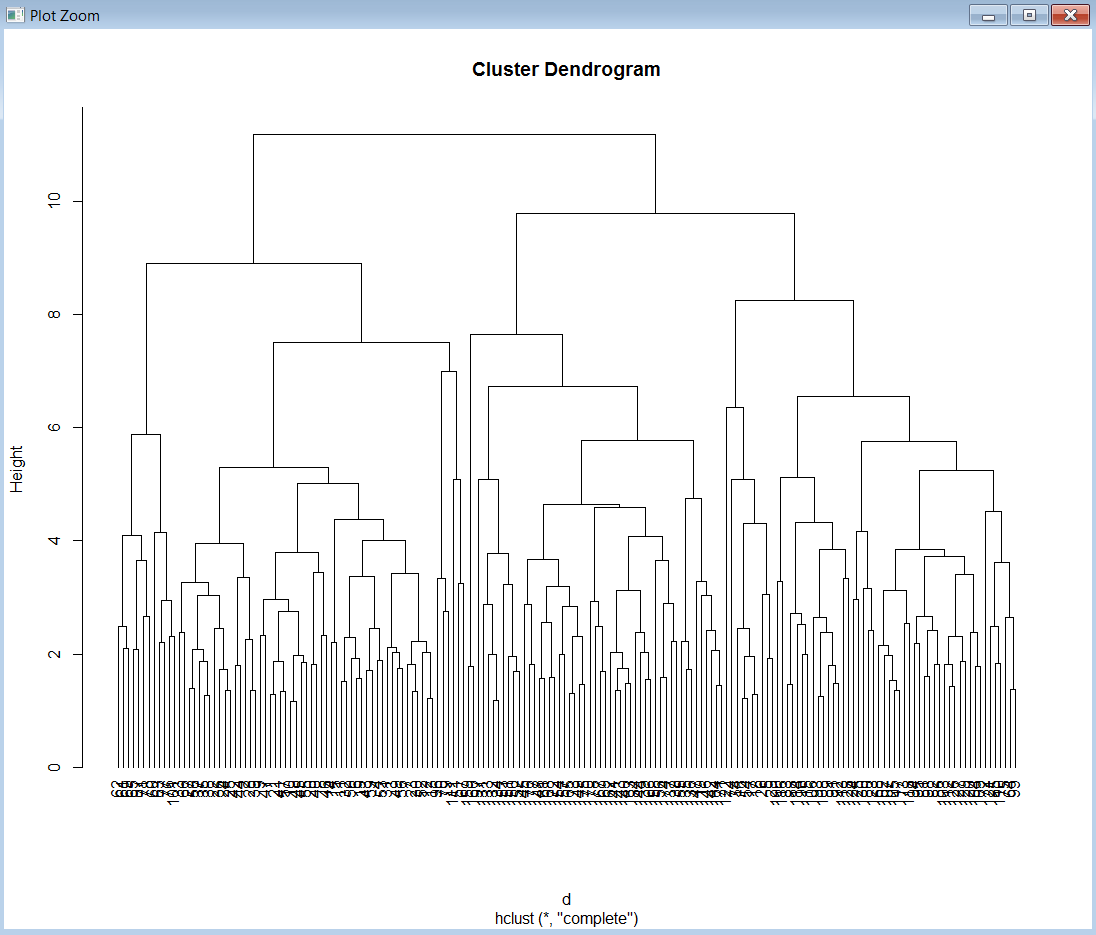
We do the normalization next



We try to apply hclust first , even though the amount of data is more

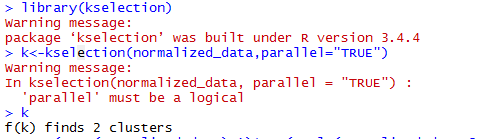


We can see that the dendogram is not clear



Hence our next action is to make use of kmeans, in order to use kmeans we need to find the number of cluster using kselection and elbow curve

With kselection the number of required cluster is

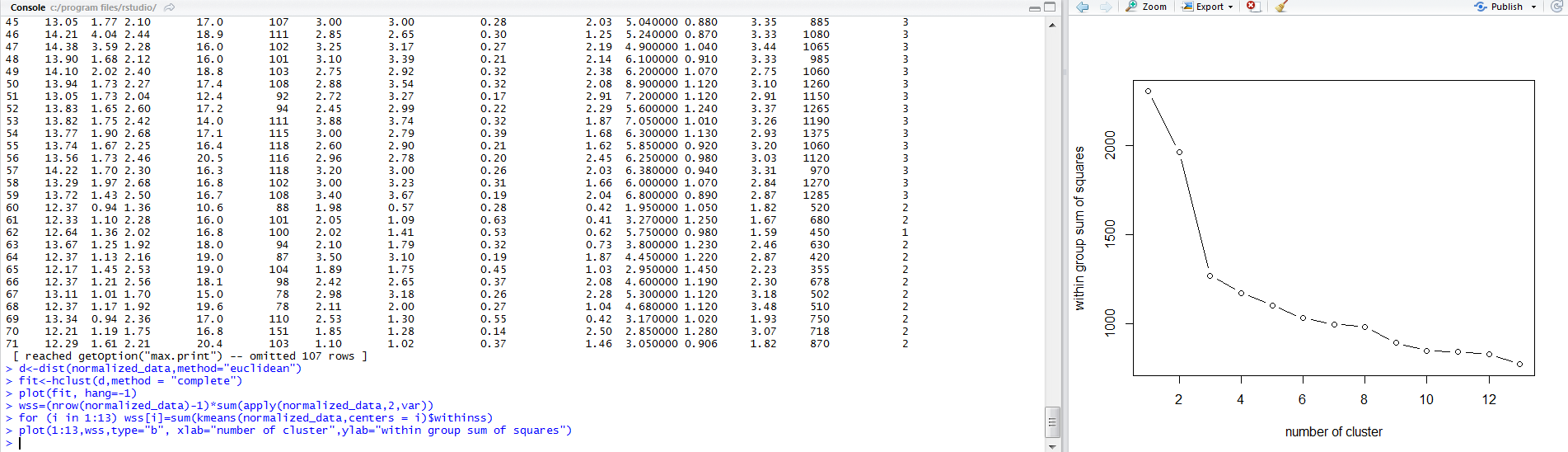


Let’s try scree plot/elbow curve

> wss=(nrow(normalized\_data)-1)\*sum(apply(normalized\_data,2,var))

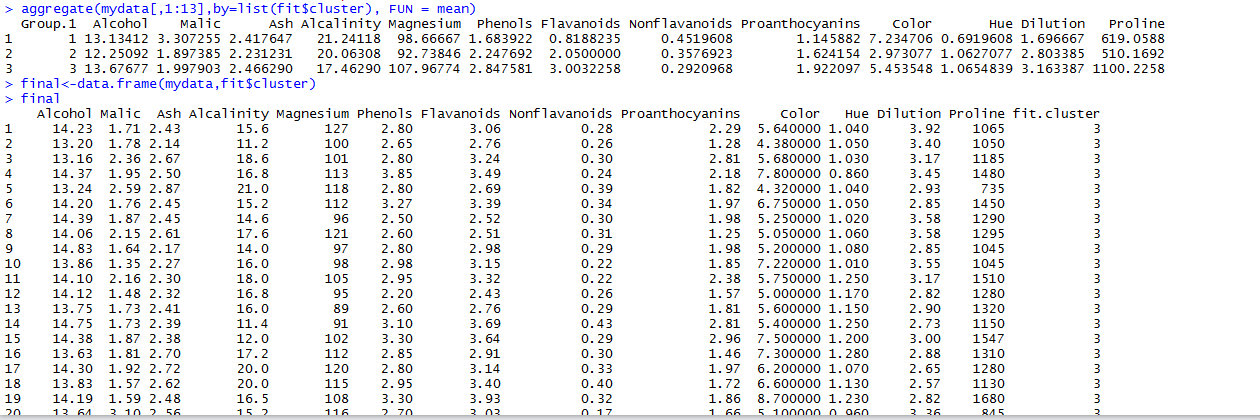
> for (i in 1:13) wss[i]=sum(kmeans(normalized\_data,centers = i)$withinss)

> plot(1:13,wss,type="b", xlab="number of cluster",ylab="within group sum of squares")

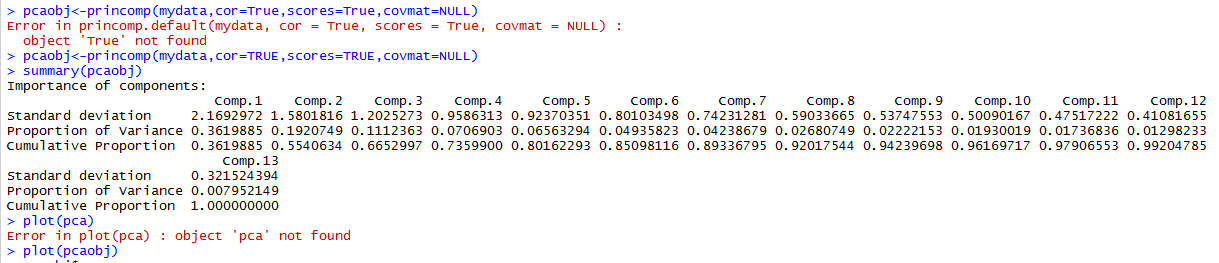


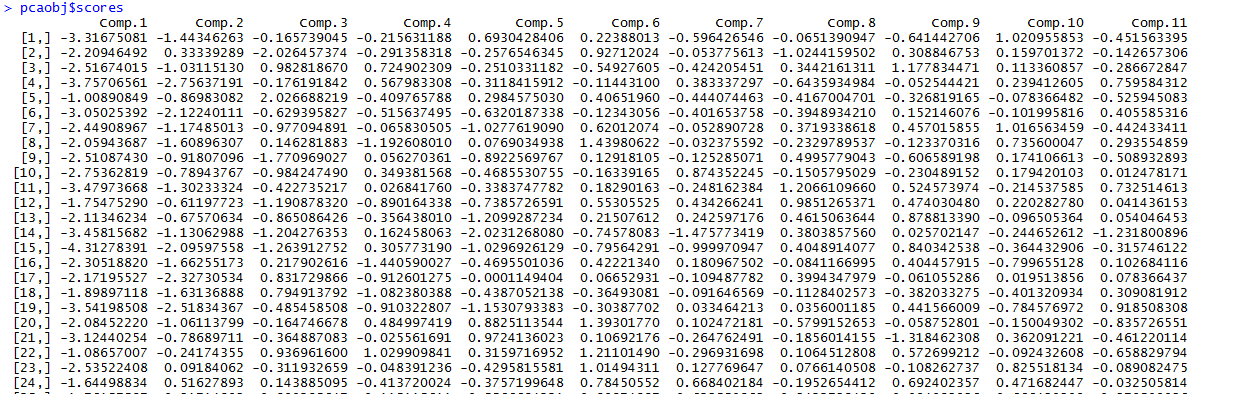
We can see an elbow at 3; hence let’s take 3 as the number of clusters required

> fit<-kmeans(normalized\_data,3)



Now let’s try applying pca on the raw data





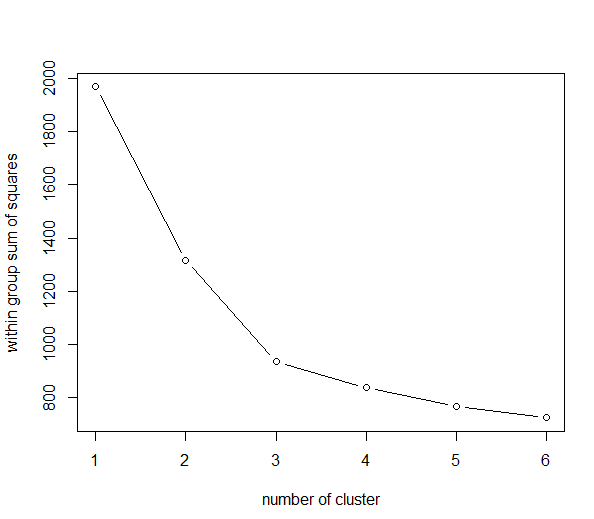
> clus\_data<-pcaobj$scores[,1:6]

> normalized\_data<-clus\_data

> wss=(nrow(normalized\_data)-1)\*sum(apply(normalized\_data,2,var))

> for (i in 1:6) wss[i]=sum(kmeans(normalized\_data,centers = i)$withinss)

> plot(1:6,wss,type="b", xlab="number of cluster",ylab="within group sum of squares")



Then we apply kmeans algorithm

> fit<-kmeans(normalized\_data,3)

> aggregate(mydata[,1:13],by=list(fit$cluster), FUN = mean)

> final<-data.frame(mydata,fit$cluster)